

오픈소스로 여는 뉴노멀

2020 공개SW 페스티벌

서울시 공공데이터를 이용한 Kibana 대시보드 만들기

Elastic
김종민

▶ Elastic? Kibana?

 elastic 은 데이터 검색과 분석을 위한

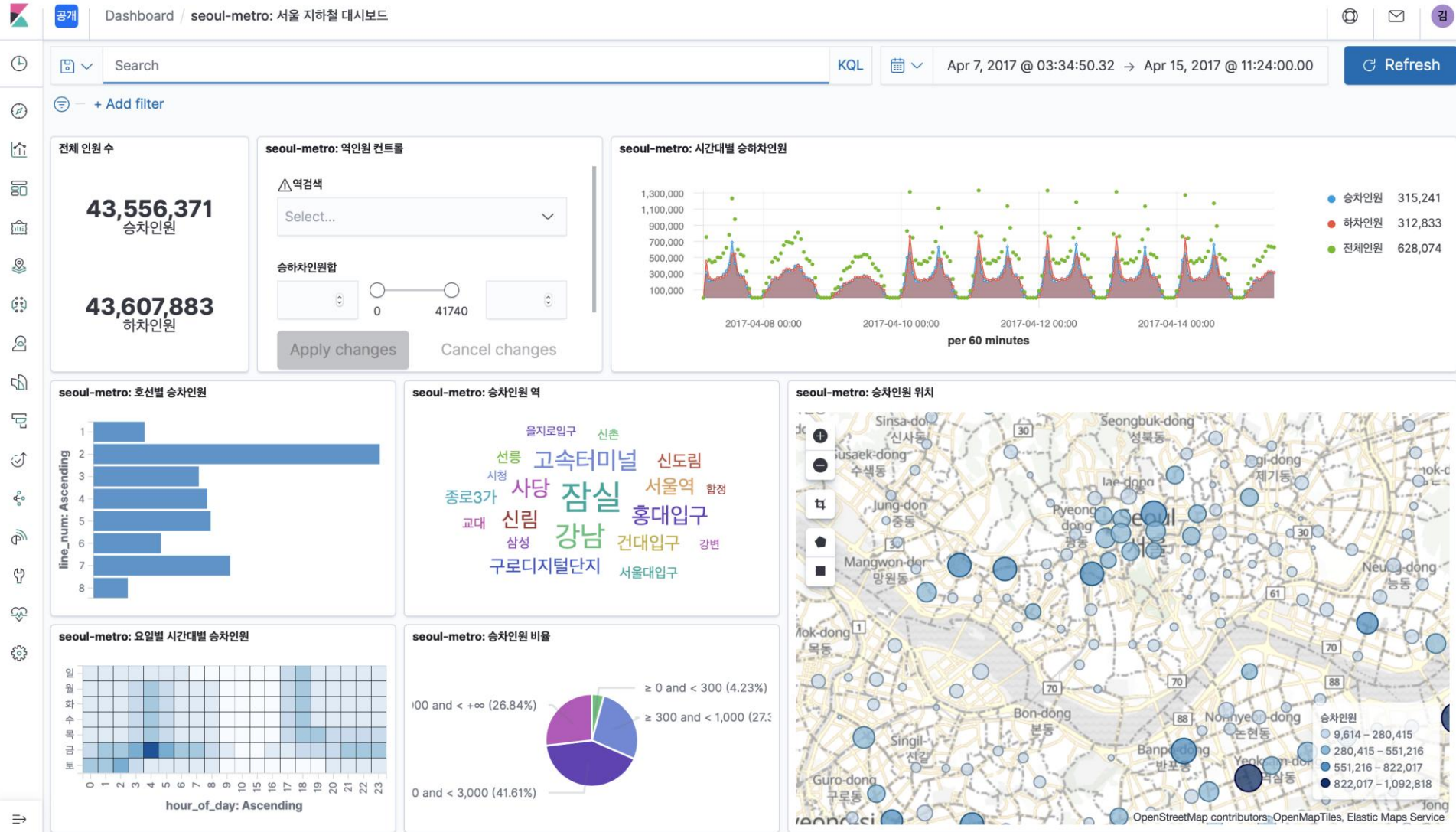
유명한 검색엔진인  **Elasticsearch** 를 비롯하여

데이터 시각화 도구인  **Kibana**

그리고 데이터 수집기인  **Beats**  **Logstash**

들을 개발중인 기업입니다.

서울 지하철 승하차 인원 대시보드



처음 계기는
2015년 10월 EMOCON
Elasticsearch + Kibana
데모

×

EMOCON 2015 F/W

📅 Fri, Oct 23, 2015 9:50 PM

<https://www.crowdcast.io/e/emocon2015>




이상한모임 온라인 컨퍼런스 2015

Elasticsearch + Kibana
 빠르게 확산되었으나
 관련 정보가 많이 없어
 서울시 지하철 데모가
 여기 저기서 학습자료로
 많이 사용되었습니다.

Google

Elastic Stack를 이용한 서울시 지하철 대시보드(2019) - 1 : 네이버 블로그
[m.blog.naver.com > PostView > Translate this page](https://m.blog.naver.com/PostView)
 Mar 21, 2019 - 이전엔 Elasticsearch + Logstash + Kibana를 묶어 ELK stack이라 불렀 ... 에 관해서는 Elastic Stack을 이용한 서울시 지하철 대시보드(2019) - 2 ...

Images for ELK 스택을 사용한 서울시 지하철 대시보드 만들기



→ More images for ELK 스택을 사용한 서울시 지하철 대시보드 만들기 Report images

네트워크 보안 엔지니어를 위한 엘라스틱 강좌 - 1강 - 오픈베이스 연구소
[https://oblab.tistory.com > ... > Translate this page](https://oblab.tistory.com)
 Jan 24, 2018 - 오늘(첫 날)은 Elastic Stack을 소개하고, 활용사례를 알려드립니다. ... EMOCON 2015 F/W ELK 스택을 사용한 서울시 지하철 대시보드 만들기.

ELK (데이터 분석도구) 활용하기 - 호학심사 심지기의 (好學深 心知基意)
[https://oog.co.kr > ... > Translate this page](https://oog.co.kr)
 Jan 18, 2016 - Kibana 는 Elasticsearch 에 저장된 Data 를 사용자에게 Dashboard 형태로 ... 2015 F/W ELK 스택을 사용한 서울시 지하철 대시보드 만들기 추천.

ELK Kibana 사용법 - kenu - Medium
[https://medium.com > elk-kibana-사용법-96c353db2bbc > Translate this page](https://medium.com)
 Feb 18, 2016 - 이 문서에서는 오픈소스 데이터 시각화 도구인 Kibana의 사용법을 설명하려고 한다. Kibana ... ELK 스택을 사용한 서울시 지하철 대시보드 만들기 ...

오픈소스 데이터 시각화 패키지 ELK(Elasticsearch + Logstash ... - O...
[https://okky.kr > article > Translate this page](https://okky.kr)
 May 23, 2016 - Elasticsearch와 Kibana는 권장 버전을 맞춰야 함; 설치 위치 /opt/ 또는 ~/local/ 권장 2016 ELK 스택으로 서울시 지하철 대시보드 만들기 추천 ... EMOCON 2015 F/W ELK 스택을 사용한 서울시 지하철 대시보드 만들기.

데이터는 서울시 공공데이터 포털 자료를 이용했습니다.

<https://data.seoul.go.kr>

모든 서울시민을 위한 공공데이터

열린데이터광장에서 서울시와 연계 기관이 공개한 공공데이터를 확인하실 수 있습니다. 서울시와 관련된 다양한 공공데이터를 확인해 보세요.

| | | |
|-------|--------|----------|
| 데이터셋 | 서비스 | Open API |
| 6,716 | 13,672 | 4,983 |

| | | | | | |
|--|---|---|---|--|--|
|  보건 |  일반행정 |  문화/관광 |  산업/경제 |  복지 |  환경 |
|  교통 |  도시관리 |  교육 |  안전 |  인구/가구 |  주택/건설 |

관심데이터
36종 생활이슈
 자세히보기 >



많은 자료들이 공개되어 있지만, 정작 잘 정리된 자료들은 많지 않았습니다.

- 주로 월 단위 통계
- 누락된 자료가 많거나
- UTF-8 이 아닌 인코딩

데이터셋

공공데이터 (4)
통계 (0)

카테고리

교통 (4)

제공유형

FILE (3)
OpenAPI (3)
SHEET (3)


관련태그 더보기

승차 (4)
지하철역 (4)
하차 (4)
교통 (3)
통계 (3)

인기검색어

1. 구별*
2. 동별*
3. 지하철*
4. 따릉이*
5. 유동인구*
6. 공공자전거*
7. 범죄*


'역별 승하차 인원'의 검색 결과 4건을 찾았습니다. 조희순 ▼ 조회



서울시 지하철호선별 역별 승하차 인원 정보

교통카드(선후불교통카드 및 1회용 교통카드)를 이용한 지하철호선별 역별(서울교통공사, 한국철도공...
수정일자: 2020-06-28 제공기관: 서울특별시 제공부서: 도시교통실 교통기획관 교통정책과


[교통] SHEET OpenAPI FILE



서울시 지하철 호선별 역별 시간대별 승하차 인원 정보

교통카드(선후불교통카드 및 1회용 교통카드)를 이용한 지하철 호선별 역별(1~9호선, 서울시 관할 ...
수정일자: 2020-03-03 제공기관: 서울특별시 제공부서: 도시교통실 교통기획관 교통정책과


[교통] SHEET OpenAPI



서울시 지하철 호선별 역별 유/무임 승하차 인원 정보

교통카드(선후불교통카드 및 1회용 교통카드)를 이용한 지하철 호선별 역별(1~9호선, 서울시 관할 ...
수정일자: 2020-02-03 제공기관: 서울특별시 제공부서: 도시교통실 교통기획관 교통정책과

[교통] SHEET OpenAPI FILE



서울교통공사 연도별 일별 시간대별 역별 승하차 인원

서울교통공사 연도별 일별 시간대별 역별 승하차인원 (2008년~2019년)
수정일자: 2020-06-02 제공기관: 서울교통공사 제공부서: IT기획처

[교통] FILE

7

그래도 나름 가장 정리가 잘 된 데이터가 서울교통공사 연도별 일별 시간대별 역별 승하차 인원



교통

공공데이터

활용갤러리 등록

URL 복사

목록 이동

서울교통공사 연도별 일별 시간대별 역별 승하차 인원

서울교통공사 연도별 일별 시간대별 역별 승하차인원 (2008년~2019년)

파일내려받기

* 파일에 이상이 있는 경우 '오류신고'를 통해 운영자에게 알려주세요.

오류신고

| NO | 항목 | 파일명 | 용량 (MB) | 수정일 | 내려받기 |
|----|-----|---|---------|------------|------|
| 1 | 데이터 | 서울교통공사 2019년 일별 역별 시간대별 승하차인원(1_8호선).xlsx | 30.2 | 2020.06.03 | |
| 2 | 데이터 | 서울교통공사 2018년 일별 역별 시간대별 승하차인원(1_8호선).xlsx | 30.1 | 2019.05.10 | |
| 3 | 데이터 | 서울교통공사 2017년 일별 역별 시간대별 승하차인원(1_8호선).xlsx | 31.8 | 2019.05.10 | |
| 4 | 데이터 | 서울교통공사 2016년 일별 역별 시간대별 승하차인원(1~8호선).csv | 29.4 | 2017.08.28 | |
| 5 | 데이터 | 서울교통공사 2015년 일별 역별 시간대별 승하차인원(1~8호선).csv | 29.3 | 2017.08.28 | |

[전체 파일보기](#)

- 2008년 부터 년도별로 구분된 Excel 파일로 다운로드 제공
- 날짜/호선/역 별로 승차/하차 2행씩 구분
- 각 열에 1시간 단위로 집계된 데이터

| | A | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P |
|----|----------|-----|-----|------|----|--------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|
| 1 | | | | | | | | | | | | | | | | |
| 2 | 날짜 | 호선 | 역번호 | 역명 | 구분 | 06시 이전 | 06 ~ 07 | 07 ~ 08 | 08 ~ 09 | 09 ~ 10 | 10 ~ 11 | 11 ~ 12 | 12 ~ 13 | 13 ~ 14 | 14 ~ 15 | 15 ~ 16 |
| 3 | 2019.1.1 | 1호선 | 150 | 서울역 | 승차 | 348 | 321 | 348 | 741 | 940 | 1401 | 2166 | 2844 | 3055 | 2341 | 3370 |
| 4 | 2019.1.1 | 1호선 | 150 | 서울역 | 하차 | 222 | 821 | 808 | 961 | 1472 | 1737 | 1917 | 2316 | 2539 | 2360 | 2743 |
| 5 | 2019.1.1 | 1호선 | 151 | 시청 | 승차 | 87 | 98 | 143 | 206 | 280 | 395 | 500 | 672 | 673 | 868 | 946 |
| 6 | 2019.1.1 | 1호선 | 151 | 시청 | 하차 | 48 | 237 | 323 | 571 | 577 | 648 | 744 | 960 | 996 | 875 | 964 |
| 7 | 2019.1.1 | 1호선 | 152 | 종각 | 승차 | 669 | 318 | 217 | 203 | 333 | 365 | 666 | 802 | 1103 | 1294 | 1843 |
| 8 | 2019.1.1 | 1호선 | 152 | 종각 | 하차 | 68 | 179 | 293 | 546 | 926 | 945 | 1334 | 1566 | 1750 | 1771 | 1654 |
| 9 | 2019.1.1 | 1호선 | 153 | 종로3가 | 승차 | 227 | 198 | 148 | 205 | 347 | 479 | 933 | 1184 | 1228 | 1636 | 1604 |
| 10 | 2019.1.1 | 1호선 | 153 | 종로3가 | 하차 | 38 | 180 | 261 | 424 | 703 | 961 | 1595 | 1977 | 2040 | 2038 | 1447 |
| 11 | 2019.1.1 | 1호선 | 154 | 종로5가 | 승차 | 48 | 80 | 67 | 138 | 207 | 362 | 686 | 991 | 1364 | 1537 | 1449 |
| 12 | 2019.1.1 | 1호선 | 154 | 종로5가 | 하차 | 25 | 102 | 140 | 296 | 509 | 763 | 974 | 1717 | 1875 | 1773 | 1413 |
| 13 | 2019.1.1 | 1호선 | 155 | 동대문 | 승차 | 138 | 117 | 159 | 259 | 429 | 468 | 563 | 654 | 754 | 1007 | 957 |
| 14 | 2019.1.1 | 1호선 | 155 | 동대문 | 하차 | 36 | 109 | 109 | 187 | 310 | 448 | 797 | 1054 | 1101 | 1077 | 908 |
| 15 | 2019.1.1 | 1호선 | 156 | 신설동 | 승차 | 143 | 144 | 155 | 217 | 349 | 454 | 612 | 538 | 531 | 561 | 561 |
| 16 | 2019.1.1 | 1호선 | 156 | 신설동 | 하차 | 37 | 126 | 131 | 152 | 241 | 275 | 378 | 503 | 501 | 569 | 475 |
| 17 | 2019.1.1 | 1호선 | 157 | 제기동 | 승차 | 272 | 168 | 153 | 220 | 324 | 441 | 542 | 792 | 978 | 1026 | 1282 |
| 18 | 2019.1.1 | 1호선 | 157 | 제기동 | 하차 | 40 | 118 | 155 | 180 | 307 | 523 | 821 | 1334 | 1520 | 1131 | 989 |

- Elasticsearch 에 저장하기 위해서는 JSON 형식이어야 합니다.
- 각 이벤트를 단일 문서로 저장해야 합니다.
- Timestamp 는 ISO8601 "**yyyy-MM-dd'T'HH:mm:ss.SSSZ**" 형식으로

```
"_source" : {  
  "@timestamp" : "2019-01-20T20:00:00.000Z",  
  "line_num" : "7호선",  
  "station" : {  
    "name" : "가산디지털단지"  
  },  
  "people" : {  
    "in" : 663,  
    "out" : 2393  
  }  
}
```

- Excel 파일을 2행씩 루프를 돌면서 승차/하차 구분
- F열 부터 시각(hour) 으로 분리해서 루프

| | A | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P |
|----|----------|-----|-----|------|----|--------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|
| 1 | | | | | | | | | | | | | | | | |
| 2 | 날짜 | 호선 | 역번호 | 역명 | 구분 | 06시 이전 | 06 ~ 07 | 07 ~ 08 | 08 ~ 09 | 09 ~ 10 | 10 ~ 11 | 11 ~ 12 | 12 ~ 13 | 13 ~ 14 | 14 ~ 15 | 15 ~ 16 |
| 3 | 2019.1.1 | 1호선 | 150 | 서울역 | 승차 | 348 | 321 | 348 | 741 | 940 | 1401 | 2166 | 2844 | 3055 | 2341 | 3370 |
| 4 | 2019.1.1 | 1호선 | 150 | 서울역 | 하차 | 222 | 821 | 808 | 961 | 1472 | 1737 | 1917 | 2316 | 2539 | 2360 | 2743 |
| 5 | 2019.1.1 | 1호선 | 151 | 시청 | 승차 | 87 | 98 | 143 | 206 | 280 | 395 | 500 | 672 | 673 | 868 | 946 |
| 6 | 2019.1.1 | 1호선 | 151 | 시청 | 하차 | 48 | 237 | 323 | 571 | 577 | 648 | 744 | 960 | 996 | 875 | 964 |
| 7 | 2019.1.1 | 1호선 | 152 | 종각 | 승차 | 669 | 318 | 217 | 203 | 333 | 365 | 666 | 802 | 1103 | 1294 | 1843 |
| 8 | 2019.1.1 | 1호선 | 152 | 종각 | 하차 | 68 | 179 | 293 | 546 | 926 | 945 | 1334 | 1566 | 1750 | 1771 | 1654 |
| 9 | 2019.1.1 | 1호선 | 153 | 종로3가 | 승차 | 227 | 198 | 148 | 205 | 347 | 479 | 933 | 1184 | 1228 | 1636 | 1604 |
| 10 | 2019.1.1 | 1호선 | 153 | 종로3가 | 하차 | 38 | 180 | 261 | 424 | 703 | 961 | 1595 | 1977 | 2040 | 2038 | 1447 |
| 11 | 2019.1.1 | 1호선 | 154 | 종로5가 | 승차 | 48 | 80 | 67 | 138 | 207 | 362 | 686 | 991 | 1364 | 1537 | 1449 |
| 12 | 2019.1.1 | 1호선 | 154 | 종로5가 | 하차 | 25 | 102 | 140 | 296 | 509 | 763 | 974 | 1717 | 1875 | 1773 | 1413 |
| 13 | 2019.1.1 | 1호선 | 155 | 동대문 | 승차 | 138 | 117 | 159 | 259 | 429 | 468 | 563 | 654 | 754 | 1007 | 957 |
| 14 | 2019.1.1 | 1호선 | 155 | 동대문 | 하차 | 36 | 109 | 109 | 187 | 310 | 448 | 797 | 1054 | 1101 | 1077 | 908 |
| 15 | 2019.1.1 | 1호선 | 156 | 신설동 | 승차 | 143 | 144 | 155 | 217 | 349 | 454 | 612 | 538 | 531 | 561 | 561 |
| 16 | 2019.1.1 | 1호선 | 156 | 신설동 | 하차 | 37 | 126 | 131 | 152 | 241 | 275 | 378 | 503 | 501 | 569 | 475 |
| 17 | 2019.1.1 | 1호선 | 157 | 제기동 | 승차 | 272 | 168 | 153 | 220 | 324 | 441 | 542 | 792 | 978 | 1026 | 1282 |
| 18 | 2019.1.1 | 1호선 | 157 | 제기동 | 하차 | 40 | 118 | 155 | 180 | 307 | 523 | 821 | 1334 | 1520 | 1131 | 989 |

- 데이터 정제를 위해 간단한 프로그램을 짰습니다.

<https://github.com/eskrug/elastic-demos>

```
// csv 파일 형태는 아래와 같이 되어야 함.
// 0      1      2      3      4      5      6      7      ... 23      24      25
// 날짜,   호선,   역번호, 역명,   구분, 05~06, 06~07, 07~08, ... 23~24, 00~01, 합계
// 2018-01-01, 1호선, 150,   서울역, 승차, 373,   318,   365,   ... 781,   96,   40393

// 2줄씩 루프 돌면서 0~3 열 까지의 데이터가 동일한지 확인
for(var cd=1; cd< csv_data.length ; cd+=2){
  var dataIn = csv_data[cd];
  var dataOut = csv_data[cd+1];
  if(dataIn[0]===dataOut[0] && dataIn[1]===dataOut[1]
    && dataIn[2]===dataOut[2] && dataIn[3]===dataOut[3]){
    // 역명
    var station_name = dataIn[3];
    // 날짜
    var ldateTemp = dataIn[0].split('-');
    // 시간 값으로 루프
    for(var h=0; h < 20; h++){
      var ldate = new Date(ldateTemp[0],Number(ldateTemp[1])-1,ldateTemp[2],h);
      // 승차인원
```

2015 년도 EMOCON 데모에는

서울교통공사 연도별 일별 시간대별 역별 승하차 인원

데이터만 활용했지만, 최근에는 유용한 데이터를 더 접목했습니다.



[교통]

서울교통공사 지하철 역명 다국어 표기 정보

1~8호선 지하철 역명에 대한 한글, 한자, 영자, 중국어, 일어 표기 정보 서비스입니다.
수정일자: 2020-06-02 제공기관: 서울교통공사 제공부서: IT기획처

FILE

FOFO 데이터

‘서울시 역코드로 지하철역 위치 조회’의 검색 결과 0건을 찾았습니다.

정확도순 ▼

조회

검색 결과가 없습니다.

- 다국어 역명 정보
- 역별 위치 정보 (위도/경도)
- 역별 시간대별 승/하차 인원 정보

```

"_source" : {
  "station" : {
    "en" : "Taereung",
    "ch" : "泰陵",
    "jp" : "テルニック",
    "name" : "태릉입구",
    "kr" : "태릉입구",
    "chc" : "泰陵入口"
  },
  "location" : {
    "lat" : 37.617983,
    "lon" : 127.07512
  },
  "line_num" : "7호선",
  "people" : {
    "total" : 84,
    "in" : 76,
    "out" : 8
  },
  "@timestamp" : "2019-02-04T15:00:00.000Z",
  "code" : "2719",
  "line_num_en" : "Line 7"
}

```

다음과 같은 Kibana 대시보드를 완성할 수 있습니다.

공개
Dashboard / seoul-metro: 서울 지하철 대시보드
김

KQL

Apr 7, 2017 @ 03:34:50.32 → Apr 15, 2017 @ 11:24:00.00

+ Add filter
Refresh

전체 인원 수

43,556,371

승차인원

43,607,883

하차인원

seoul-metro: 역인원 컨트롤

역검색

Select...

승하차인원합

0 41740

Apply changes
Cancel changes

seoul-metro: 시간대별 승하차인원

- 승차인원 315,241
- 하차인원 312,833
- 전체인원 628,074

seoul-metro: 호선별 승차인원

seoul-metro: 승차인원 위치

seoul-metro: 승차인원 위치

seoul-metro: 승차인원 위치

seoul-metro: 승차인원 비율

seoul-metro: 요일별 시간대별 승차인원

전체 인원 수

141,396,564
승차인원

141,429,852
하차인원

seoul-metro: 역인원 컨트롤

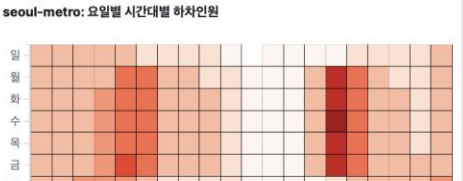
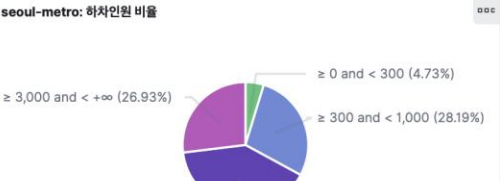
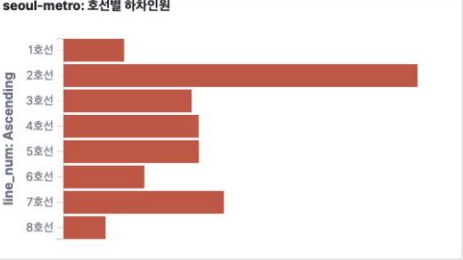
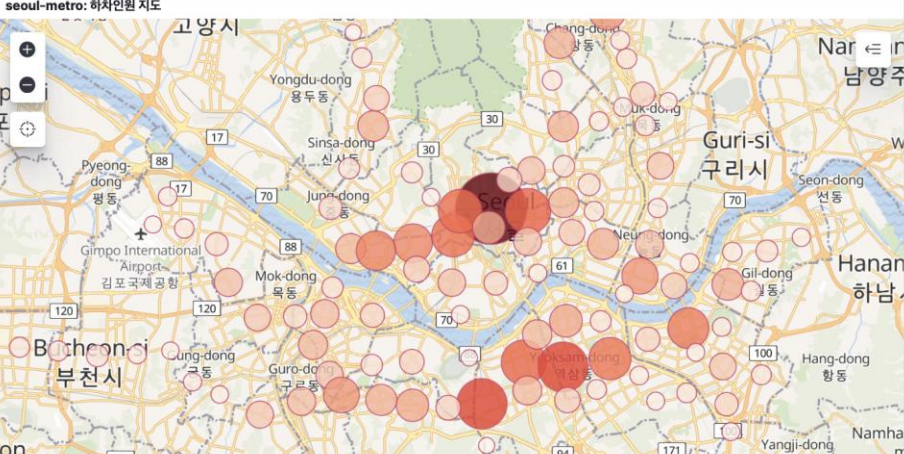
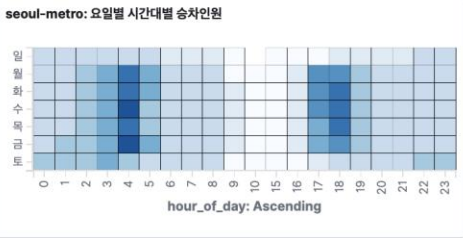
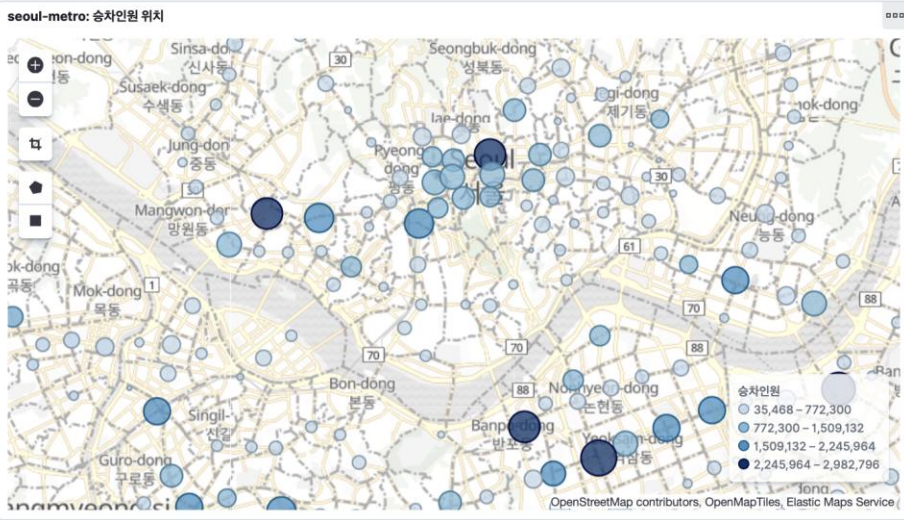
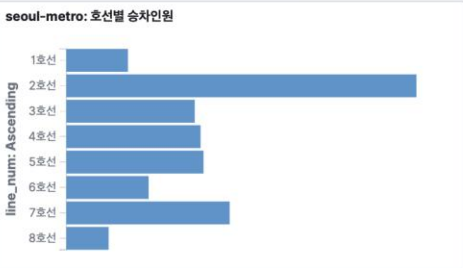
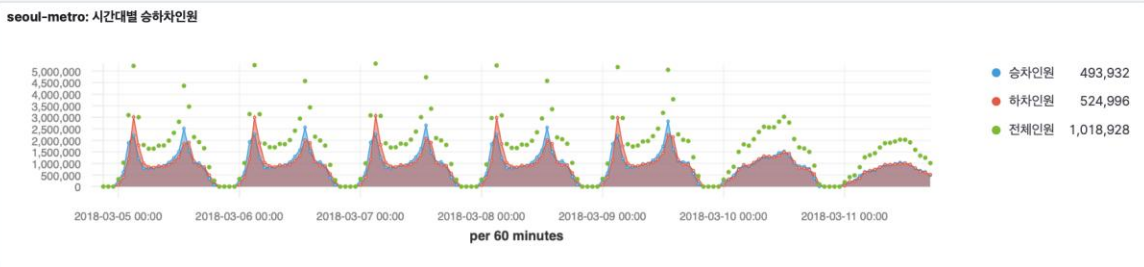
△역검색

Select...

승차인원합

0 41740

Apply changes Cancel changes



전체 인원 수

10,604,092
승차인원

11,159,552
하차인원

seoul-metro: 역인원 컨트롤

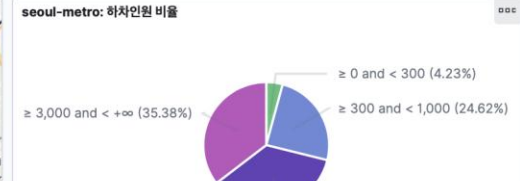
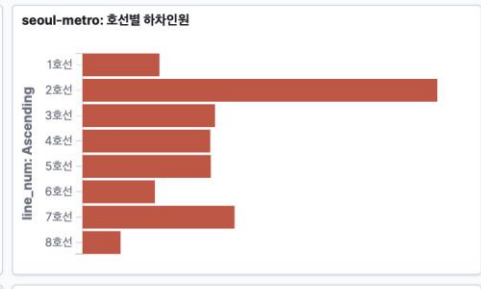
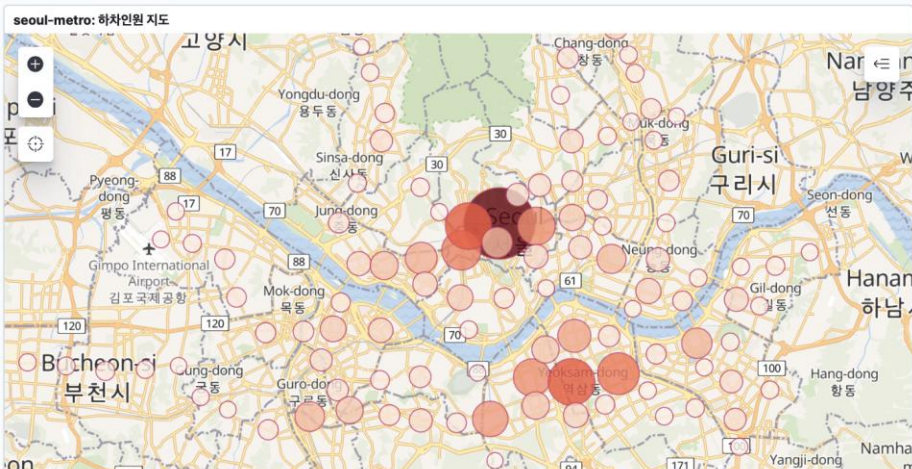
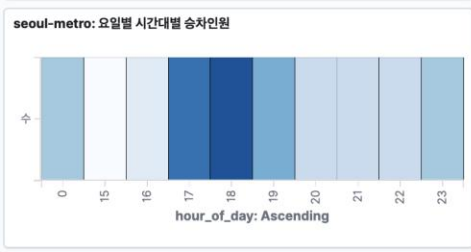
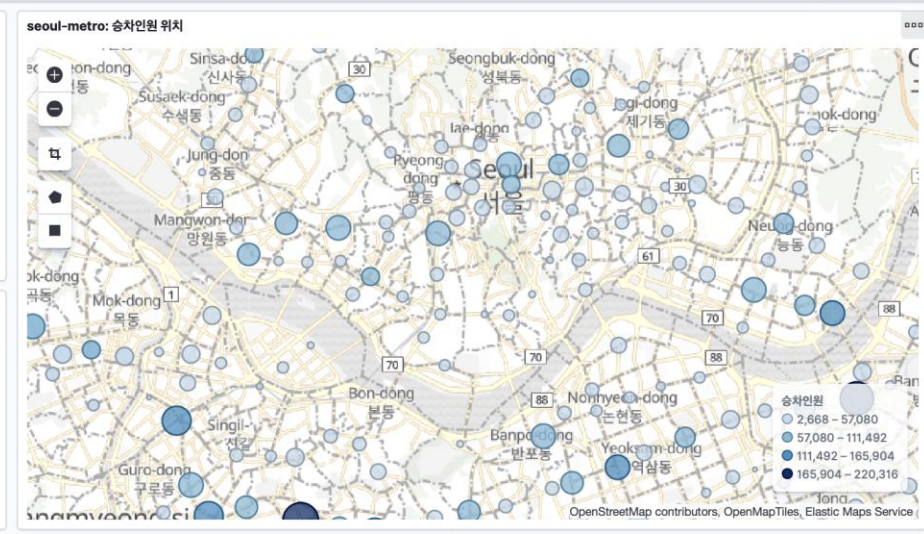
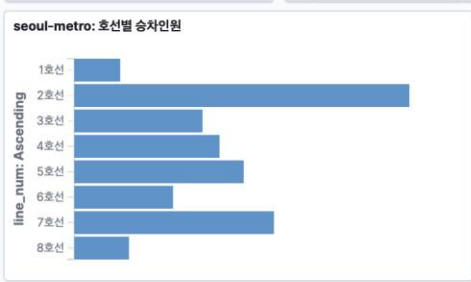
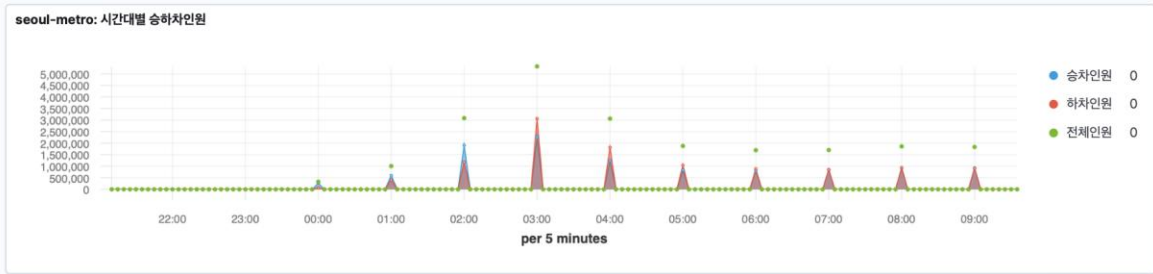
△역검색

Select...

승하차인원함

0 41740

Apply changes Cancel changes



전체 인원 수

7,650,244
승차인원

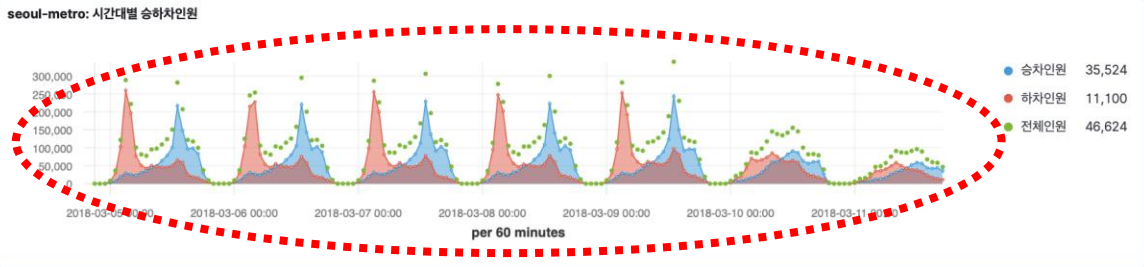
7,645,752
하차인원

seoul-metro: 역인원 컨트롤

역검색
Select...

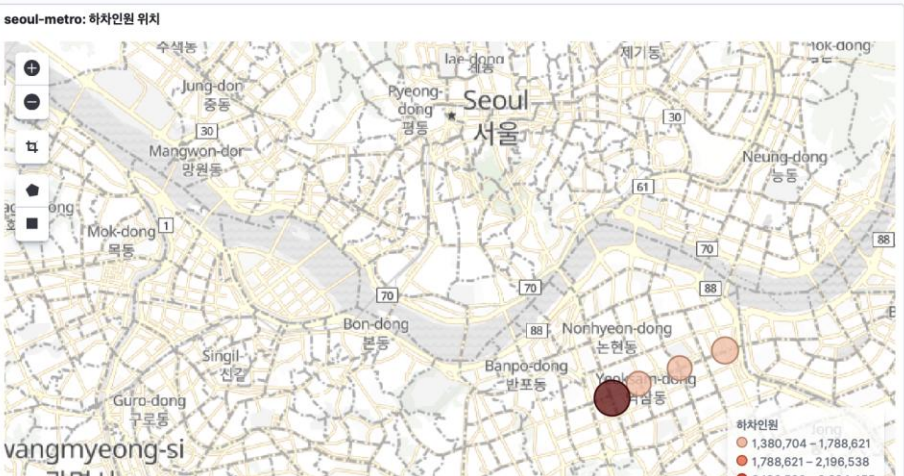
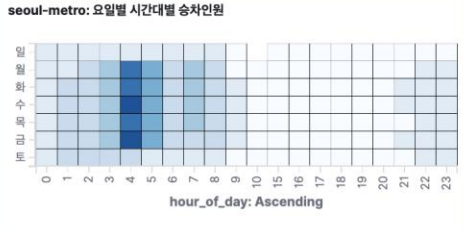
승하차인원합
0 41740

Apply changes Cancel changes



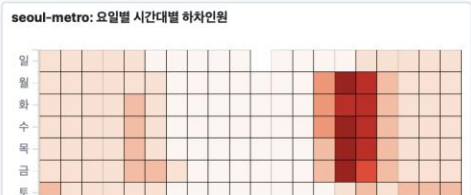
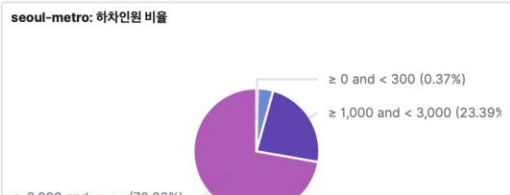
seoul-metro: 승차인원 역

선릉 역삼 강남 삼성
삼성(무역센터)



seoul-metro: 하차인원 역

역삼 강남 삼성
선릉 삼성(무역센터)



전체 인원 수

34,109,852
승차인원

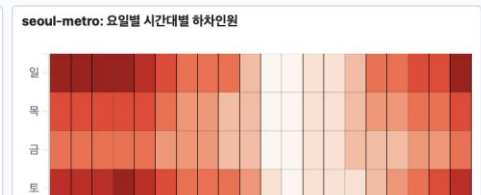
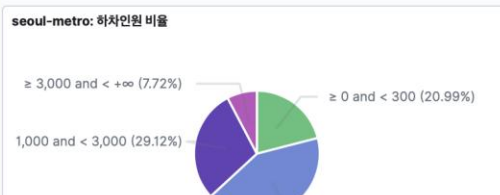
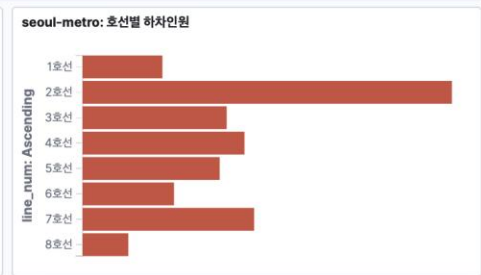
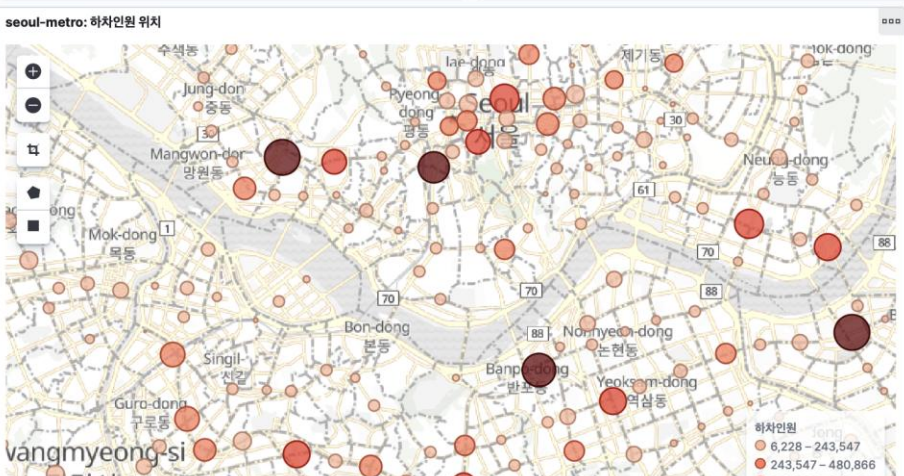
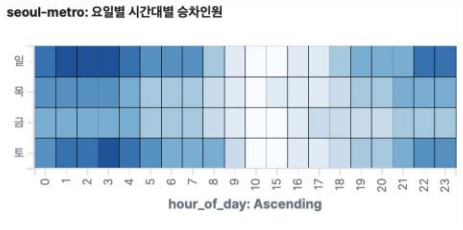
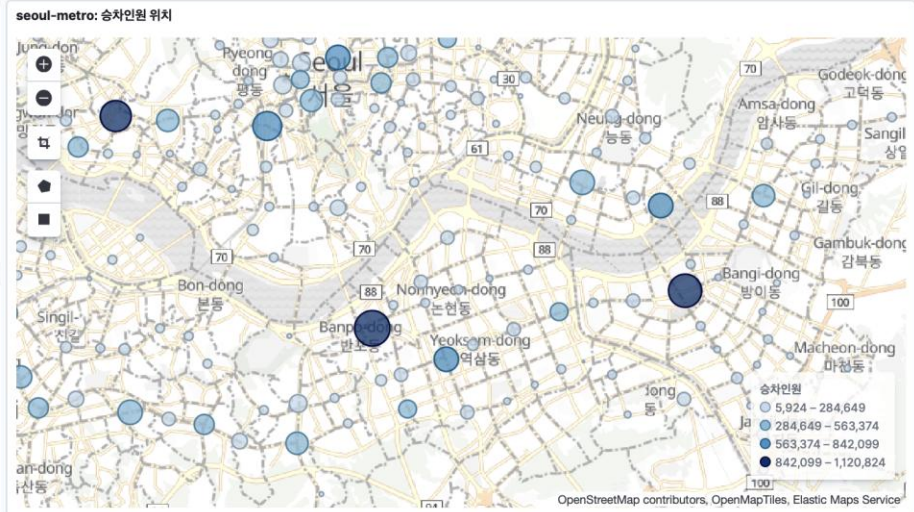
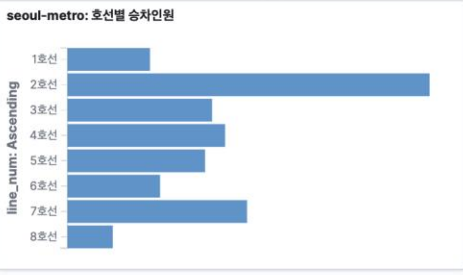
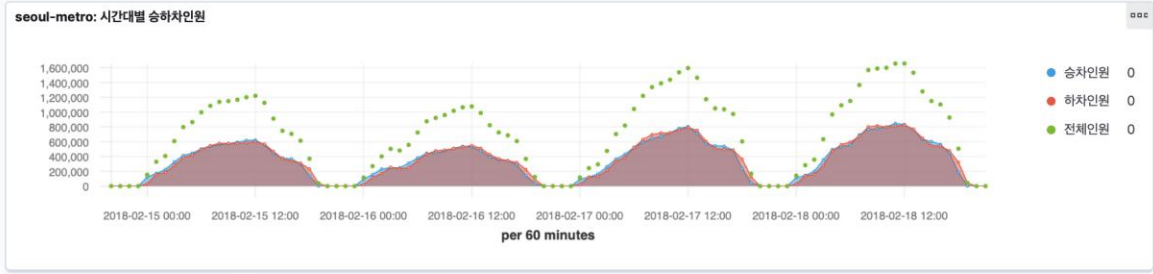
34,051,752
하차인원

seoul-metro:역인원 컨트롤

△역검색
Select...

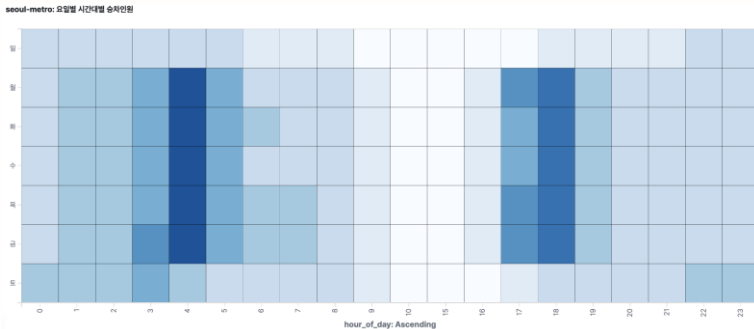
승하차인원합
0 41740

Apply changes Cancel changes



요일별 시각별 그래프를 위해 새 필드 저장

- hour_of_day
- day_of_week



```

| | | | "source" : ""def ts=ctx['@timestamp'];
def sdf=new SimpleDateFormat("yyyy-MM-dd'T'HH:mm:ss.SSS");
def date=sdf.parse(ts);
def cal=Calendar.getInstance();
cal.setTime(date);

ctx.hour_of_day=cal.get(Calendar.HOUR_OF_DAY);

def dowNum=cal.get(Calendar.DAY_OF_WEEK)-1;
def dowEn=["Sun", "Mon", "Tue", "Wed", "Thu", "Fri", "Sat"][dowNum];
def dowKr=["일", "월", "화", "수", "목", "금", "토"][dowNum];

ctx.day_of_week=["num":dowNum, "en":dowEn, "kr":dowKr];""

```

```

"_source": {
  "code": "2553",
  "line_num": "5호선",
  "people": {
    "total": 717,
    "in": 171,
    "out": 546
  },
},
"line_num_en": "Line 5",
"@timestamp": "2018-12-31T19:00:00.000+09:00",
"station": {
  "ch": "高德",
  "jp": "コドク",
  "kr": "고덕",
  "name": "고덕",
  "en": "Godeok",
  "chc": "高德"
},
},
"location": {
  "lon": 127.154151,
  "lat": 37.555004
},
"hour_of_day": 19,
"day_of_week": {
  "num": 1,
  "kr": "월",
  "en": "Mon"
}
},

```

공공데이터를 제공하는 기업/기관에게 부탁드립니다.

- .xlsx 대신 .csv .json
- EUC-KR 대신 UTF-8
- 열 대신 행
- 지하철 역별 위치정보는 왜 없어졌을까? □

<https://github.com/eskrug/elastic-demos>

공공 데이터 수집

서울 열린 데이터 광장 접속 : <https://data.seoul.go.kr/>

지하철역 위치 정보.

검색어 : 서울시 역코드로 지하철역 위치 조회

- json 파일로 다운로드
- `source/station_info.json` 으로 저장

다국어 역명 정보

검색어 : 서울교통공사 지하철 역명 다국어 표기 정보

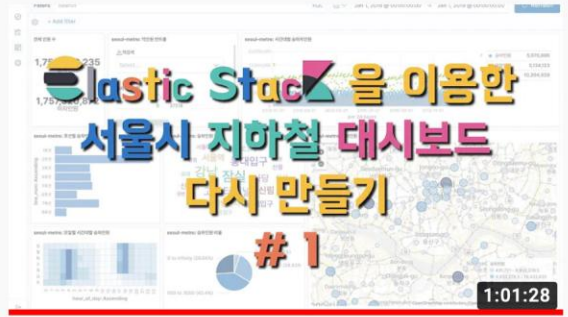
- json 파일로 다운로드
- `source/station_lang.json` 으로 저장

역별 승하차 인원 데이터셋

검색어 : 서울교통공사 연도별 일별 시간대별 역별 승하차 인원

- 2018년 파일 다운로드
- Excel 로 열어서 편집 -> bin/run.js 파일을 주석을 참고해서
 - 1 행 삭제
 - B 열 (구분) 삭제
 - 다른 이름으로 저장 --> `source/metro_log_2018.csv`
- `source/metro_log_2018.csv` 파일 인코딩 EUC-KR 에서 UTF-8 로 변경 후 저장

공통 필터



Elastic Stack을 이용한 서울시 지하철 대시보드 다시 만들기 #1

조회수 4.4천회 · 1년 전

🇰🇷 한국 Elastic 사용자 그룹

2015년에 만들어 인기를 끌었던 "ELK 스택을 사용한 서울시 지하철 대시보드 만들기" 를 최신 버전의 Elastic Stack 을 이용해서 다시 만들어 봅니다. #1 영상에서는 다음 내용들을 이야기합니다. - 서울시 공공데이터 포털에서 데이터 수집 - 역 위치정보 + 다국어 역명 정보 + 역별 시간대



Elastic Stack을 이용한 서울시 지하철 대시보드 다시 만들기 #2

조회수 1.1천회 · 1년 전

🇰🇷 한국 Elastic 사용자 그룹

2015년에 만들어 인기를 끌었던 "ELK 스택을 사용한 서울시 지하철 대시보드 만들기" 를 최신 버전의 Elastic Stack 을 이용해서 다시 만들어 봅니다. #2 영상에서는 다음 내용들을 이야기합니다. - Kibana 스페이스 생성 및 전환 - Kibana 인덱스 패턴 생성 - Discover : 필드 선택 및 쿼리



Elastic Stack을 이용한 서울시 지하철 대시보드 다시 만들기 #3

조회수 1.2천회 · 1년 전

🇰🇷 한국 Elastic 사용자 그룹

2015년에 만들어 인기를 끌었던 "ELK 스택을 사용한 서울시 지하철 대시보드 만들기" 를 최신 버전의 Elastic Stack 을 이용해서 다시 만들어 봅니다. #3 영상에서는 다음 내용들을 이야기합니다. - Painless 스크립트 언어 - 스크립트 필드 - Ingest Pipeline 인제스트 파이프라인 -

thank
you!